(12) UK Patent Application (19) GB (11) 2 356 523 (13) A

(43) Date of A Publication 23.05.2001

(21) Application No 9927500.0

(22) Date of Filing 19.11.1999

(71) Applicant(s)
Mitel Corporation
(Incorporated in Canada - Ontario)
PO Box 13089, 350 Legget Drive, Kanata, Ontario, K2K 2W7, Canada

(72) Inventor(s)
Brian Cheng
Oliver Yang
Fujian Li

(74) Agent and/or Address for Service
Beresford & Co
2-5 Warwick Court, High Holborn, LONDON, WC1R 5DH, United Kingdom

(51) INT CL[7]
H04L 12/56 // H04L 29/06 , H04Q 11/04

(52) UK CL (Edition S)
H4K KTKX

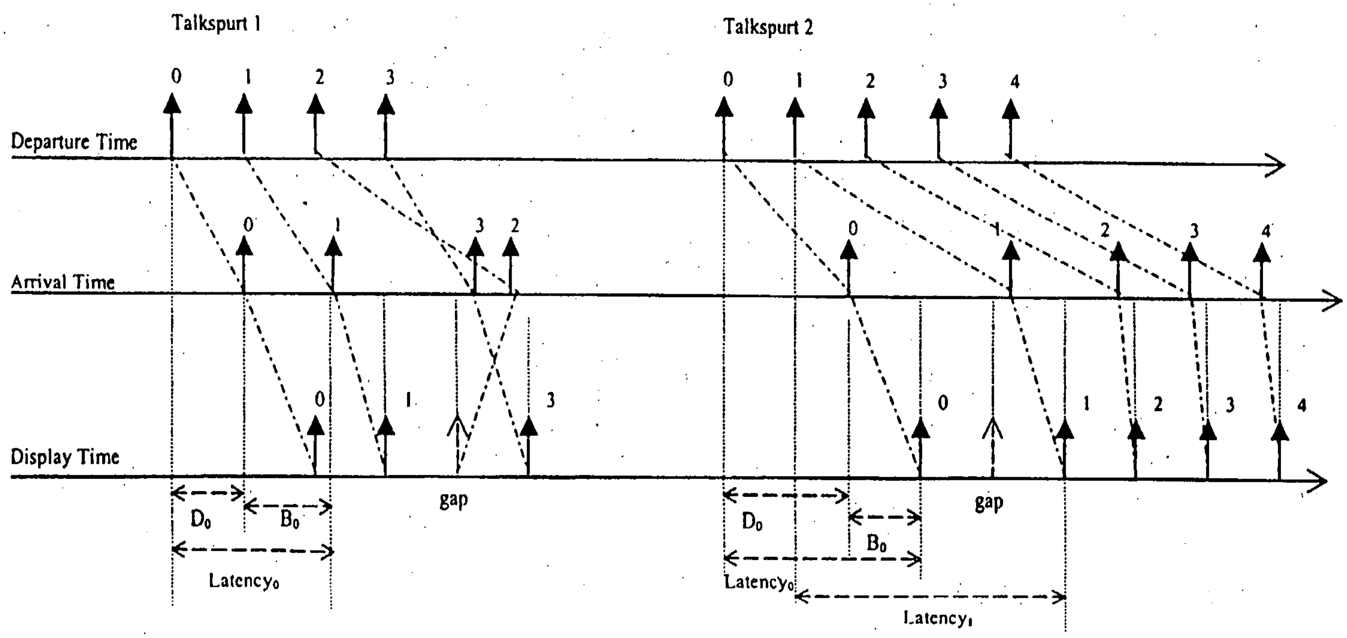
(56) Documents Cited
None

(58) Field of Search
UK CL (Edition R) H4K KTK
INT CL[7] H04L 12/56 29/06 , H04Q 11/04
ONLINE : WPI ; EPODOC ; JAPIO

(54) Abstract Title
Dynamic adaptation voice audio jitter control method

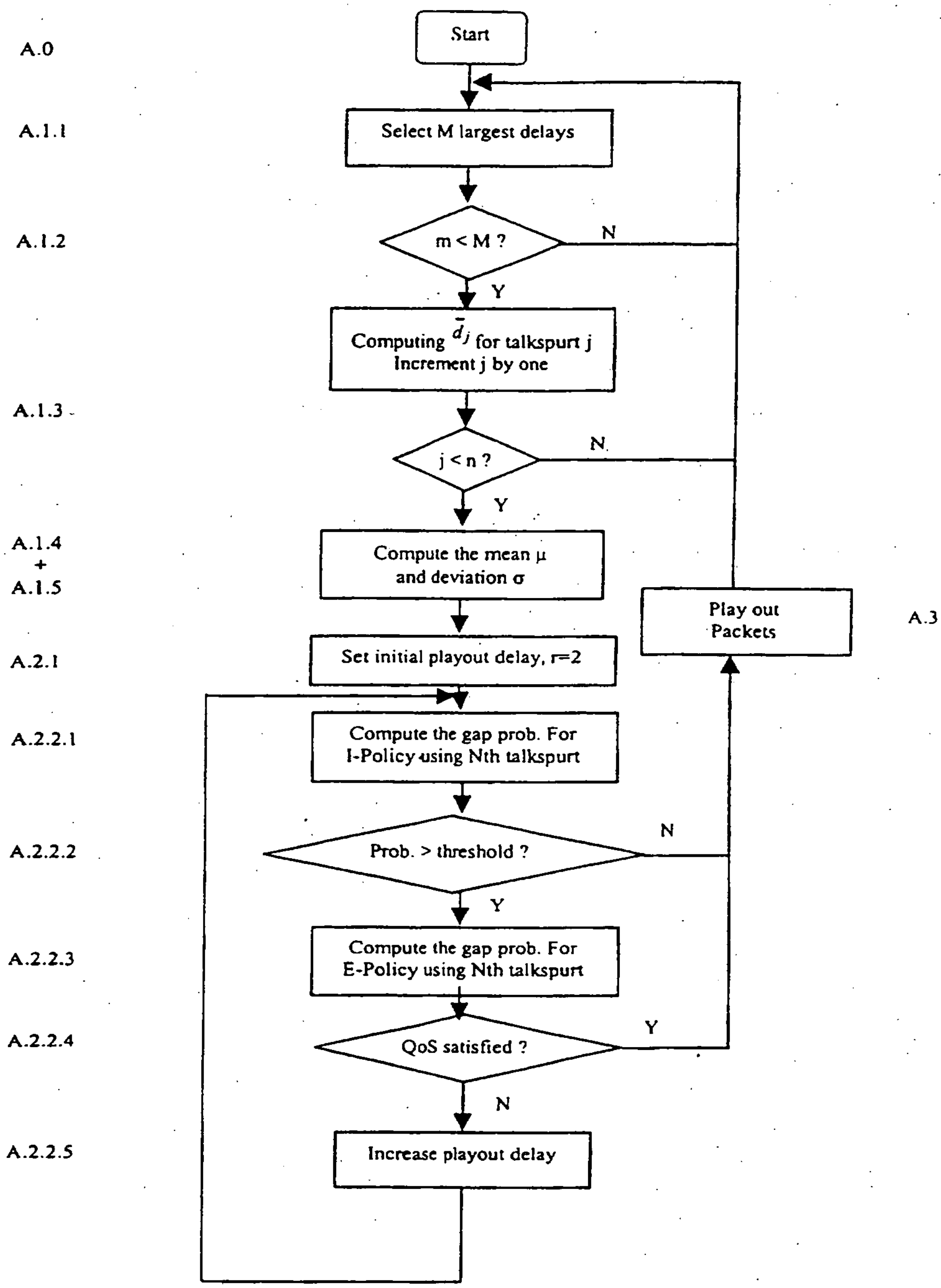
(57) A jitter control system is provided for determining display latency and simultaneously reducing audio gaps in received voice packets. Three methods are set forth for estimating the mean delay and delay variation for the voice packets from collected measurements. Based on these estimations, a Gaussian delay distribution is built for the packet arrival. Then, probabilities for single gap and multiple contiguous gaps for a given display delay are calculated and the packet is played out. Hence, the display delay can be dynamically adjusted according to recent delay characteristics and tradeoff policy between gap frequency and display latency.

GB 2 356 523 A



Talkspurt 1
Talkspurt 2
Departure Time
Arrival Time
Display Time
gap
gap
D_0
B_0
$\mathrm{Latency}_0$
D_0
B_0
$\mathrm{Latency}_0$
$\mathrm{Latency}_1$

FIGURE 1



A.0
Start
A.1.1
Select M largest delays
A.1.2
m < M ?
N
Y
Computing $\bar{d}_j$ for talkspurt j
Increment j by one
A.1.3
j < n ?
N
Y
A.1.4
+
A.1.5
Compute the mean μ
and deviation σ
Play out
Packets
A.3
A.2.1
Set initial playout delay, r=2
A.2.2.1
Compute the gap prob. For
I-Policy using Nth talkspurt
A.2.2.2
Prob. > threshold ?
N
Y
A.2.2.3
Compute the gap prob. For
E-Policy using Nth talkspurt
A.2.2.4
QoS satisfied ?
Y
N
A.2.2.5
Increase playout delay

FIGURE 2

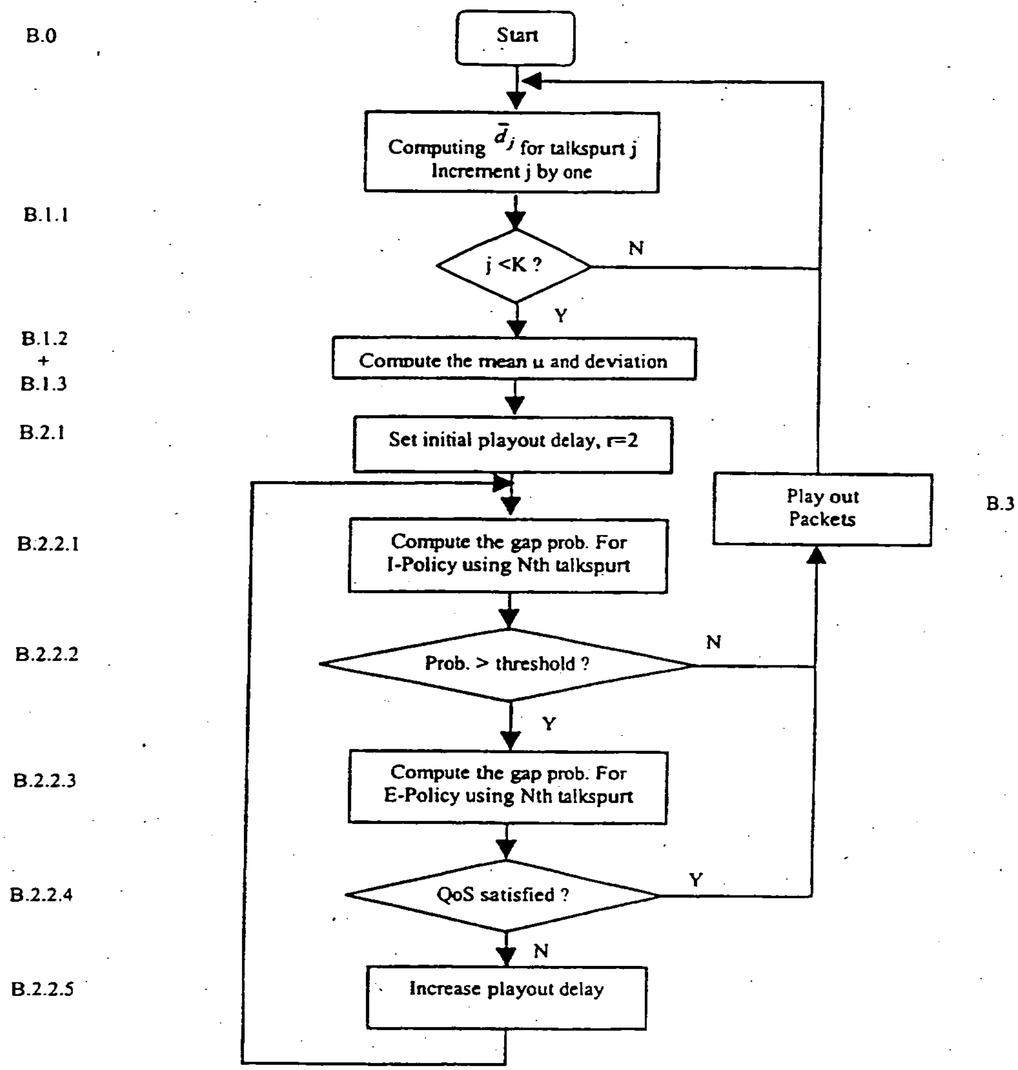

B.0
Start
Computing $\bar{d}_j$ for talkspurt j
Increment j by one
B.1.1
j <K ?
N
Y
B.1.2
+
B.1.3
Compute the mean u and deviation
B.2.1
Set initial playout delay, r=2
Play out
Packets
B.3
B.2.2.1
Compute the gap prob. For
I-Policy using Nth talkspurt
B.2.2.2
Prob. > threshold ?
N
Y
B.2.2.3
Compute the gap prob. For
E-Policy using Nth talkspurt
B.2.2.4
QoS satisfied ?
Y
N
B.2.2.5
Increase playout delay

FIGURE 3

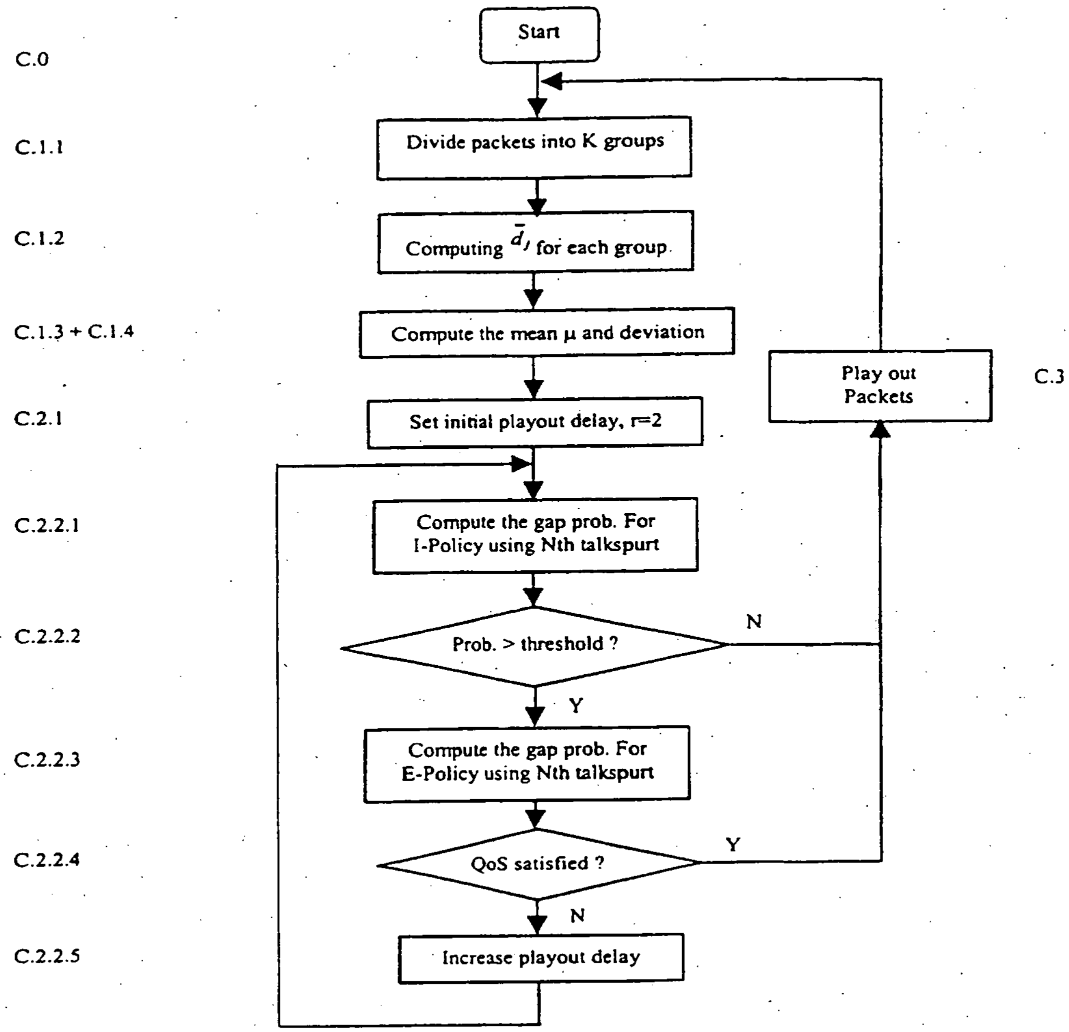

C.0
Start
C.1.1
Divide packets into K groups
C.1.2
Computing $\bar{d}_j$ for each group
C.1.3 + C.1.4
Compute the mean μ and deviation
C.2.1
Set initial playout delay, r=2
C.2.2.1
Compute the gap prob. For I-Policy using Nth talkspurt
C.2.2.2
Prob. > threshold ?
N
Y
C.2.2.3
Compute the gap prob. For E-Policy using Nth talkspurt
C.2.2.4
QoS satisfied ?
Y
N
C.2.2.5
Increase playout delay
Play out Packets
C.3

FIGURE 4

2356523

DYNAMIC ADAPTATION VOICE AUDIO JITTER CONTROL METHOD

FIELD OF THE INVENTION

This invention relates in general to communication systems and more specifically to an apparatus and method for estimating jitter in voice packets received over a network running the Internet Protocol (IP).

BACKGROUND OF THE INVENTION

According to known techniques for network transmission of voice audio over the Internet, a voice signal is sampled, compressed and packetized, and then transmitted over the network. However, as a result of stochastic unpredictable end-to-end network delays, the network introduces some delay and delay variation (jitter) to each delivered packet. It is a requirement of such network voice transmission systems that the voice signal be faithfully recovered via depacketizing, decompressing, and buffering in such a way as to minimize voice packet delay variations which otherwise would result in potentially unintelligible voice transmissions.

Buffering of voice packets before playback has been used to address the problem of jitter in networked voice communications, but gives rise to audio playout delay. Playout delay is a primary factor in determining the perceived quality and interactivity of voice communication across networks. If the worst-case end-to-end delay (e.g. 150ms) is chosen as the playout delay, there will be no gaps, but the large one-way delay makes

normal human conversation extremely difficult. In the presence of such a transmission delay, participants in a conversation are inclined to begin talking at the same time based on auditory cues resulting from perceived pauses in conversation which are, in fact, a result of the transmission delay. Normal conversation cannot be sustained in these circumstances. Voice playout with low playout latency and some gaps is preferable to high latency and no gaps. Therefore, a tradeoff is often made between playout latency and gaps in order to maintain voice quality in a network-based conversation.

Thus, one approach is to playout audio packets with a latency less than the worst-case end-to-end delay but with an acceptable gap probability. Intuitively, higher latency results in lower gap probability. However, for a given playout latency, sometimes packets waiting in the buffer will need to be discarded in order to guarantee a fixed playout delay. A lower playout delay results in a higher discard probability.

Current Internet transmission protocols provide only a "best effort" quality of service (QoS). This means that QoS guarantees for real-time audio traffic are usually accomplished outside of the network. Playback buffer management systems are well known in the art for jitter control. According to such systems network delay behavior is predicted by "mining" the most recent network packet delay patterns. Patterns are recognized by observing queuing behavior of packets in the voice playback buffer. Unfortunately, no adequate model has yet been devised for

characterizing or predicting stochastic network conditions with high reliability. The main hurdle to identifying such a model is in developing methods to adaptively respond to unknown network delay changes by dynamically adjusting the playout delay. The following prior art has been published concerning this problem:

[1] J. Bolot, "End-to-end packet delay and loss behavior in the Internet, " SIGCOMM'93, pp.289-298.

[2] D. Sanghi, O. Gudmundsso and A. Agrawala, "Experimental assessment of end-to-end behavior on Internet," Proc. IEEE Infocom'93, pp. 867-874.

[3] A. Privalov and K. Sohraby, "Per-stream jitter analysis in CBR ATM multiplexers," IEEE/ACM Trans. on Networking, Vol.6 No.2, April 1998.

[4] D. Ferrari, "Delay jitter control scheme for packet-switching internetworks," Technical Report, University of California and International Computer Science Institute, Berkeley, 1992.

[5] R. Ansari and A. R. Kaye, "Compressed voice in integrated services frame relay networks: voice synchronization," Research Report, Dept. of Systems and Comput. Engineering, Carleton University.

[6] L. Dong and A. R. Kaye "Transmission of compressed voice over integrated services frame relay networks: priority service and adaptive buildout delay", IEE Proc. Commun., 14 (4): 265-274, Aug. 1994.

[7] W. E.Naylor and L. Kleinrock, "Stream traffic communication in packet-switched networks: destination buffering considerations, " IEEE Trans. on Commun. Vol. 30, Dec., 1982, pp. 2527-2534.

[8] Po L. Tien and Maria C. Yuang, "Intelligent voice smoother for VBR voice over ATM networks," Infocom'98

[9] Po L. Tien and Maria C. Yuang, "Intelligent video smoother for multimedia communications, " IEEE JSAC. Vol.15, Feb. 1997, pp. 136-146

[10] R. Ramjee, Jim Kurose, Don, Towsley and Henning Schulzrinne, "Adaptive playout mechanisms for framized audio applications in wide-area networks, " Infocom'94, pp.680-688.

[11] H. Schulzrinne, "Voice communication across the Internet: A network voice terminal, ", Technical Report TR92-50, Dept. of Computer Science, MIT, Amherst, Massachusetts, July 1992.

[12] Tom Xu, Jitter Management Research Report, Technical Report, Multimedia Communication Research Lab, University of Ottawa.

According to the prior art Buffer Monitoring Method (see references [5] and [7], above), the holding time for buffering received packets is used as a measure of the network congestion level. Smaller holding times represent higher network delays, because received packets are played out faster than incoming new packets. Changes in holding times are used to adjust playout length or

latency. If the change exceeds a predetermined threshold, the playout length is expanded or compressed in small amounts. Another approach to buffer monitoring relates to queue length. The queue length is measured at each packet display initiation time which decreases the queuing packets by one. A threshold is established for each queue length for a given lifetime or duration. If the queue length is found to be greater than the given threshold for the lifetime period, incoming (newest) or oldest audio packets are discarded depending on the threshold value; higher threshold value means preferably that incoming packets have to be dropped, rather than the oldest one. This has the effect of shortening the playout latency (or playback queue) for new voice packets, without causing an audio gap (silence). Another extension of this method determines display latency based on observed delays at the beginning of each talk spurt of which the length depends very much on the individuals and the subject of the conversation. According to this method only "m" recent delays are recorded. The display latency is chosen for a given talk spurt after discarding the "k" largest delays according to defined rules for choosing "m" and "k" as parameters. For a given latency, one of two jitter management policies may be applied, referred to in the art as I-policy or E-policy. According to the I-policy, audio packets are displayed with a fixed display latency, and each late audio packet (i.e. a packet with end-to-end delay greater than this latency) will be discarded. According to the E-policy, late audio

packets are always displayed at the next possible opportunity, even those that would disrupt the normal voice communication.

According to the second prior art method, referred to as Predetermined Delay Distribution Method (see references [8] and [9], above), the threshold is analytically computed and the playout delay is determined in advance based on the predetermined delay distribution. The playout rate is reduced if packets in the playout buffer fall below a given threshold. The main drawback of this method is that it may lead to the misjudgment of playout rates if the actual packet arrival process fails to follow the predetermined delay process.

Finally, according to the prior art Delay Estimation Method (see reference [10], above), the playout time of the first packet in any talkspurt is determined according to the measurements of recently-collected delay statistics. Other subsequent voice packets in the talkspurt can then be easily calculated.

SUMMARY OF THE INVENTION

The inventors have discovered that by knowing the delay distribution, it is possible to manage the tradeoff between playout latency and gaps. That is, given the playout delay, the gap probability can be determined. Given the gap probability, the playout delay can be adapted to satisfy the requirement of the gap probability. Therefore, according to the present invention, a jitter control system is provided for

determining display latency and simultaneously reducing audio gaps in received voice packets. More particularly, three methods are set forth for estimating the mean delay and delay variation for the voice packets from collected measurements. Based on these estimations, a Gaussian delay distribution is built for the packet arrival. Then, probabilities for single gap and multiple contiguous gaps for a given display delay are calculated and the packet is played out. Hence, the display delay can be dynamically adjusted according to recent delay characteristics and tradeoff policy between gap frequency and display latency.

BRIEF DESCRIPTION OF THE DRAWINGS

Embodiments of the present invention are hereinafter described with reference to the following drawings in which:

Figure 1 is a diagrammatic representation of a plurality of talkspurts each comprising a plurality of voice packets which are subject to delay;

Figure 2 is a flowchart of a jitter control algorithm according to a first embodiment of the invention;

Figure 3 is a flowchart of a jitter control algorithm according to a second embodiment of the invention; and

Figure 4 is a flowchart of a jitter control algorithm according to a third embodiment of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

In the description of embodiments of the invention appearing below, the following notations will be used:

d_0	the packet delay of the first packet in a talkspurt.
d_i	the packet delay of the i^{th} packet in a talkspurt.
$\overline{d}_j$	the mean packet delay of the m selected delay samples in a talkspurt.
D	the amount to delay the playout of the first packet in a talkspurt.
m	the number of the selected (largest) delay samples in a talkspurt.
m_j	the total number of delay samples in talkspurt j.
M	the number of samples in a sample group.
n	the number of talkspurts used to predict the Gaussian distribution in the first embodiment of the present invention.
N	total range of delay samples.
$\Pr(gap_1^I)$	the probability that a single gap occurs in a talkspurt for I-policy.
$\Pr(gap_2^I)$	the probability that a double-gap occurs in a talkspurt for I-policy.
$\Pr(gap_M^I)$	the probability that more than two gaps occur contiguously in a talkspurt for I-policy.

$\Pr(gap_1^E)$ the probability that a single gap occurs in a talkspurt for E-policy.

$\Pr(gap_2^E)$ the probability that a double-gap occurs in a talkspurt for E-policy.

$\Pr(gap_M^E)$ the probability that more than two gaps occur contiguously in a talkspurt for E-policy.

μ, $\hat{\mu}$ the mean of Gaussian distribution.

σ, $\hat{\sigma}$ the standard deviation of the derived Gaussian distribution.

Firstly, three measurement methods are defined for estimating delay and delay variation, and for deriving delay distribution from these estimations, according to the present invention. The delay distribution is then used to predict each packet's delay within a talkspurt, and to adjust playout delay for the talkspurt accordingly. The delay for each packet is computed at the receiver end provided that the first packet in a talkspurt is time-stamped, as is commonly done, and the interval between two consecutive packets in the same talkspurt is known. According to prior art reference [7], this delay calculation can be done without requirement of synchronization between source clock and destination clock.

Figure 1 shows a representative packet delay distribution for the three inventive methods set forth

in greater detail below. In Figure 1, D_0 is the network delay of the 1st packet in a talkspurt, and B_0 is the play-out delay of 1st packet. Talkspurt 1 shows the playout of packets using I-Policy; while talkspurt 2 shows the playout of packets using E-Policy. The initial value of the 1st packet playout delay is determined based on the mean μ and deviation σ of a number of previous passing packets, the thresholds of gap probabilities as well QoS requirements, as discussed in greater detail below. When the gap probabilities for I-Policy meet the threshold, the packets of current talkspurt are displayed using I-Policy with an initial value of playout delay. Alternatively, if the gap-probabilities for E-Policy meet the threshold, the packets are displayed using E-Policy with an initial value of playout delay. Otherwise, the value of the 1st packet play-out delay is increased until one threshold is met to display the next talkspurt with an appropriate policy.

The differences among the three methods according to the present invention relate to different approaches for computing μ and σ from the specific number of previous passing packets.

Assume that there are a large number of delay samples, x_i, i = 1, 2, ..., N, which can be divided into N/M groups with each group consisting of M samples. A set of N/M new variables, $\bar{x}_i$, is thus obtained

$$\overline{x}_j = \sum_{i=(j-1)N+1}^{jN} \frac{x_i}{N} \qquad (1)$$

where N>>M.

As long as M is large (e.g. more than 10), the new variable $\bar{x}$, becomes a Gaussian variable with pdf (probability density function) being expressed as

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad (2)$$

The three methods by which Gaussian delay distribution is calculated according to the present invention, are set forth below, based upon calculating the average delay of the collected samples.

According to the first inventive method, assume that the j^{th} talkspurt consists of m_j packets and let d_i, i = 1, 2, ..., m_j, be the packet delay of i^{th} packet in the talkspurt. To ensure accuracy of the delay distribution with its Gaussian parameters to be characterized, a sufficient number of talkspurts must be obtained in which there are a sufficient number of samples. Only the m largest delays out of the m_j packets' delays are recorded. If m_j<m in any j^{th} talkspurt, the talkspurt is dropped from consideration. Thus, packet delays in the dropped talkspurt are not included in the estimation of the Gaussian parameters. For each selected talkspurt, the mean packet delay can be calculated:

$$\bar{d}_j = \frac{\sum_{i=1}^{m} d_i}{m} \qquad (3)$$

For a series of n talkspurts, a set of variable values $\bar{d}_j$, j = 1, 2, ..., n, can be obtained from which the resulting Gaussian delay distribution is estimated with mean

$$\hat{\mu} = \frac{\sum_{j=1}^{n} \bar{d}_j}{n} \qquad (4)$$

and variance

$$\hat{\sigma}^2 = \sum_{j=1}^{n} (\bar{d}_j - \mu)^2 / n \qquad (5)$$

This estimation process is dynamic in the sense that the delay changes in the previous "n talkspurts" are used to predict the traffic characteristics in the "$(n+1)^{th}$ talkspurt". When audio packets in "n^{th} talkspurt" are played out, the delay changes from the "2nd talkspurt" to the "$(n+1)^{th}$ talkspurt" are used to predict the delay nature in "$(n+2)^{th}$ talkspurt" and the playout delay is adjusted accordingly.

In this method, m and n correspond to M and N/M in equation (1), respectively. Obviously, m times n is equivalent to N.

According to the second method embodied by this invention, the boundary of talkspurts to estimate the Gaussian parameters is ignored. The total range of N packet delays samples may come from different talkspurts. The delay samples are divided into N/M groups with each group having M samples. Correlation may exist between different packets that come from the same talkspurt. The Gaussian delay distribution is obtained by calculating μ and σ with equations (1) and (3)- (5) (replacing $\overline{d_i}$ with $\overline{x_i}$) under the assumption of independence among packet delays (see reference[7]). The estimation process is also dynamic in that the total range of the most recent N samples is used to estimate the delay nature of the next M packets and adjust the playout delay according to the new estimation. The difference between this method and the first method discussed above is that the delay samples used to calculate the mean for a sample group by equation (3) may belong to several consecutive talkspurts.

According to the third method of the present invention, if each talkspurt consists of a large number of packets, the samples within one talkspurt are used to estimate μ and σ by dividing the total N samples into groups each of which contains M samples. The advantage of this method is that the playout delay to be adjusted is determined by the most recent delay changes because the predicted traffic characteristics in the new talkspurt is fully based on that of the current

talkspurt. However, the estimation accuracy of this method depends on the length of the talkspurt.

After deriving the delay distributions according to one of the three methods set forth above, the gap probabilities are calculated. As discussed above D represents the playout delay of the first packet in a talkspurt. By delaying the output of the first packet in a talkspurt by an amount D, the gap frequency can be reduced in the playout. D is chosen for a talkspurt on the basis of the measurements for the previous n talkspurts. The value d_i is used to indicate the end-to-end delay of the "i^{th} packet" in a talkspurt, and d_0 is the end-to-end delay of the first packet in the talkspurt. The value of d_i may be assumed to be independently and identically distributed with the pdf given by equation (2), in which μ and σ are given by (4) and (5) respectively.

Next, the probability that a single gap or contiguous multiple gaps occurs during playout of packets is computed for the I-policy. The following analysis is based on the first method for measuring delay distribution, discussed above.

According to the I-policy, the end-to-end delay of the first packet in the talkspurt plus the playout delay is the maximum tolerance delay for each packet. The first packet in a talkspurt has to wait in the buffer for a period of D during which time the arrived packets are buffered. They are then played out at a constant rate. Each subsequent packet may experience a buffering

delay after its end-to-end delay. If a packet arrives at the receiver after the time at which its predecessor has completed emptying, a single gap will result. Hence, the gap probability is given by

$$\Pr(gap_1^I) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D) \tag{6}$$

where M represents the total number of the selected packet delays in a talkspurt and Pr(x>y) is given by

$$\Pr(x > y) = 1 - \frac{1}{\sqrt{2\pi}\sigma} \int_0^y e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{7}$$

The probability of multiple gaps occurring in a talkspurt is calculated based on the assumption that packet delays are independent each other.

For I-policy, the probability that two gaps occur contiguously is equal to the probability that the delays for a packet and its consecutive packet are both greater than that of the first packet in a talkspurt plus playout delay. For any two consecutive packets, the probability that they might create gaps is

$$\Pr(gap_2^I) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D) \tag{8}$$

The probability that more than two gaps occur contiguously is derived as

$$\Pr(gap_M^I) = \lim_{M\to\infty} \sum_{j=2}^{M-2} \frac{1}{M-j} \sum_{i=1}^{M-j} \prod_{l=i}^{i+j-1} \Pr(d_l > d_0 + D) \tag{9}$$

Under the E-policy, late packets are played out at the next opportunity instead of being discarded. This causes the playout of all packets after the late packet to be delayed. In the following derivation for E-policy, the definition for D is the same as in I-policy.

For E-policy, a gap will occur at the "i^{th} packet" if and only if

$d_i > d_0 + D$ and $d_i > d_1, d_2, \ldots, d_{i-1}$, i.e., $\prod_{j=0}^{i-1} \left(d_i > (d_j + \delta_{0j} D)\right)$,

where $\delta_{oj} = \begin{cases} 1 & \text{if } j = 0 \\ 0 & \text{otherwise} \end{cases}$

So the probability that a single gap will occur is given by

$$\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D) \tag{10}$$

Assuming independence among the delays of consecutive packets, the probabilities that a gap will occur at the "i^{th} packet" and "$(i+1)^{th}$ packet" are given respectively by

$$\Pr(\text{gap at } i) = \Pr(d_i > d_0 + D) \cdot \Pr(d_i > d_1) \cdot \Pr(d_i > d_2) \cdots \Pr(d_i > d_{i-1}) \tag{11}$$

and

$$\Pr(gap \text{ at } i+1) = \Pr(d_{i+1} > d_0 + D) \cdot \Pr(d_{i+1} > d_1) \cdot$$

$$\Pr(d_{i+1} > d_2) \cdots \Pr(d_{i+1} > d_{i-1}) \cdot \Pr(d_{i+1} > d_i) \qquad (12)$$

The probability that two gaps occur contiguously is given by

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D) \right)$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D) \right) \qquad (13)$$

The probability that more than two gaps occur contiguously is given by

$$\Pr(gap_M^E) = \begin{cases} \lim_{K\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \left(\prod_{j=0}^{i-1} Pr(d_i > d_j + d_{0j} D) \right) \left(\prod_{j=0}^{i} Pr(d_{i+1} > d_j + d_{0j} D) \right) \\ \qquad \cdots \left(\prod_{j=0}^{i+l-2} Pr(d_{i+l} > d_j + d_{0j} D) \right), \quad l = 1, 2, \ldots, M-2 \end{cases} \qquad (14)$$

As discussed briefly above, the three delay estimation methods set forth above are incorporated in this invention form implementing respective adaptive playout delay adjustment algorithms. In describing these algorithms it is assumed that the single gap probability and multiple gap probability are given, and what is needed is to adjust the playout delay in order to satisfy the QoS requirement.

The first jitter control algorithm or method is represented by the flowchart of Figure 2, for getting a

talk-spurt of m packets to play out, with M as the minimum sample size for each packet. The method begins at step A.0, following which an estimate of the mean and variance of packet delay is computed. Specifically, the following steps are executed:

A.1.1: Choose m largest delays from all the packets of each talkspurt.

A.1.2: If m is less than M, a given minimum sample size for each talkspurt, then ignore this talkspurt. Otherwise, go to A.1.3.

A.1.3: Compute the mean delay $\overline{d}_j$ of talkspurt j, j=1, 2, ..., n using equation (3).

A.1.4: Compute the mean delay μ of a series of n talkspurts by equation (4).

A.1.5: Compute the standard deviation σ of the n talkspurts by equation (5).

Next the playout delay D is determined, according to the following steps:

A.2.1: Choose the initial value of playout delay as $r*\sigma$ ($r = 2$ by default).

A.2.2: Calculate gap probability using the packet delays in the n^{th} talkspurt.

A.2.2.1: Compute the gap probability for I-policy by (6) and (8).

A.2.2.2: If the value computed by A.2.2.1 is greater than the given threshold, then go to A.2.2.3 to try E-policy. Otherwise, go to A.3.

A.2.2.3: Compute the gap probability for E-policy by (10) and (13).

A.2.2.4: If the QoS requirement is satisfied, go to A.3. Otherwise, add σ to D (step A.2.2.5) and go to A.2.2.1.

The packets of the next talkspurt are then played out (step A.3), and the algorithm returns to step A.1.1 to determine the playout delay in the next talkspurt.

The second jitter control algorithm or method is represented by the flowchart of Figure 3, for getting a talk-spurt of M packets to play out, with K as the number of talkspurts (or groups). The method begins at step B.0, following which an estimate of the mean and variance of packet delay is computed according to the second method set forth above. Specifically, the following steps are executed:

B.1.1: Compute mean delay $\bar{d}_j$ of talkspurt j, j=1, 2, ..., K, using (3) and K = N/M groups.

B.1.2: Compute the mean delay μ of a series of K groups by (4).

B.1.3: Compute the standard deviation σ for the K groups by (5).

Next the playout delay D is determined for the next talkspurt, according to the following steps:

B.2.1: Choose the initial value of playout delay as $r*\sigma$, e.g., r = 2.

B.2.2: Calculate gap probability using the packet delays in the K^{th} group and the new packet delays.

B.2.2.1: Compute the gap probability for I-policy by (6) and (8).

B.2.2.2: If the value computed by B.2.2.1 is greater than the given threshold, then go to B.2.2.3 to try E-policy. Otherwise, go to B.3.

B.2.2.3: Compute the gap probability for E-policy by (10) and (13).

B.2.2.4: If satisfy the QoS requirement, go to B.3. Otherwise, add σ to D (step B.2.2.5) and go to B.2.2.1.

The arriving packets are then played out (step B.3), and provided that the number of newly arriving packets equals M, the algorithm returns to step B.1.1.

The third jitter control algorithm or method is represented by the flowchart of Figure 4, for getting a talk-spurt of m packets to play out. The method starts at step C.0, following which an estimate of the mean and variance of packet delay is computed according to the third method set forth above. Specifically, the following steps are executed:

C.1.1: Divide the delays into K groups with each size greater than M.

C.1.2: Compute mean delay $\overline{d}_j$ of talkspurt j, j=1, 2, ..., K, by equ (3).

C.1.3: Compute the mean delay μ of the K mean values by (4).

C.1.4: Compute the standard deviation σ for the K groups of delays by (5).

Next the playout delay D is determined for the next talkspurt, according to the following steps:

C.2.1: Choose the initial value of playout delay as $r*\sigma$, e.g., $r = 2$.

C.2.2: Calculate gap probability using the packet delays in the current talkspurt.

C.2.2.1: Compute the gap probability for I-policy by (6) and (8).

C.2.2.2: If the value computed by C.2.2.1 is greater than the given threshold, then go to C.2.2.3 to try E-policy. Otherwise, go to C.3.

C.2.2.3: Compute the gap probability for E-policy by (10) and (13).

C.2.2.4: If the QoS requirement is satisfied, go to C.3. Otherwise, add σ to D (step C.2.2.5) and go to C.2.2.1.

The packets of the next talkspurt are then played out (step C.3), and at the end of the current talkspurt playout the algorithm returns to step C.1.1.

In conclusion, a method is provided for quantitatively trading off playout delay and gap probability to achieve optimum conversational performance. Given the gap probability as the quality of service, the method the amount of playout delay needed to smooth out the packet streams and computes the gap

probability for the determined playout delay. In addition, the method is applicable to either I-policy or E-policy according to specific network conditions. Thus, in the event that delay variation is not severe, I-policy can be adopted to playout the packets with a satisfactory gap probability. Otherwise, E-policy may be adopted.

The present invention can be implemented in specialised hardware. More conveniently the present invention can be implemented as a computer program on a general purpose computer. The present invention can thus be embodied as a computer program. The computer program can be provided on any carrier medium to the general purpose computer e.g. by a storage medium such as a floppy disk, CD ROM, tape, or programmable memory device or by a signal carrying the computer program such as over a network e.g. the Internet.

It will be appreciated that, although a particular embodiment of the invention has been described and illustrated in detail, various changes and modifications may be made, all of which are believed to be within the sphere and scope of the invention as defined by the claims appended hereto.

CLAIMS:

1. An adaptive playout method for voice packets received over a network, comprising the steps of:

estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets;

setting a playout delay value based on said Gaussian delay distribution;

determining probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value as required to satisfy the determined probabilities for single gap and multiple contiguous gaps between said voice packets; and

playing out said voice packets according to said playout delay.

2. The method of claim 1, wherein said step of estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets further comprises the steps of:

A.1.1: Choosing m largest delays from all voice packets of each talkspurt;

A.1.2: If m is less than a predetermined sample size M for each talkspurt then ignoring said given talkspurt, and otherwise executing step A.1.3;

A.1.3: Computing the mean delay $\bar{d}_j$ of a talkspurt j, j=1, 2, ..., n as $$\overline{d}_j = \frac{\sum_{i=1}^{m} d_i}{m};$$

A.1.4: Computing the mean delay μ of a series of n talkspurts as $$\hat{\mu} = \frac{\sum_{j=1}^{n} \overline{d}_j}{n};$$

A.1.5: Computing the standard deviation σ of said n talkspurts as $$\hat{\sigma}^2 = \sum_{j=1}^{n}(\overline{d}_j - \mu)^2 / n.$$

3. The method of claim 2, wherein said steps of setting a playout delay value based on said Gaussian delay distribution, determining said probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value further comprise the steps of:

A.2.1: Choosing an initial value D of said playout delay as r*σ;

A.2.2.1: Computing the gap probability using the packet delays in an n^{th} talkspurt for I-policy according to $$\Pr(gap_1^I) = \lim_{M \to \infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D)$$ and

$$\Pr(gap_2^I) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D)$$

A.2.2.2: If the gap probability computed at step A.2.2.1 is greater than said predetermined threshold, then executing step A.2.2.3 and otherwise executing said step of playing out said voice packets according to said playout delay;

A.2.2.3: Computing the gap probability for E-policy according to $\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)$ and

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D) \right) \cdot$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D) \right)$$

A.2.2.4: If the gap probability is less than said predetermined threshold then executing said step of playing out said voice packets according to said playout delay and otherwise adding σ to D and executing step A.2.2.1.

4. The method of claim 1, wherein said step of estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets further comprises the steps of:

B.1.1: Computing mean delay $\bar{d}_j$ of talkspurt j, j=1, 2, ..., K, using $\overline{d}_j = \frac{\sum_{i=1}^{m} d_i}{m}$ and K = N/M groups;

B.1.2: Computing the mean delay μ of a series of K groups by $\hat{\mu} = \frac{\sum_{i=1}^{n} \overline{d}_j}{n}$;

B.1.3: Computing the standard deviation σ for the K groups by $\hat{\sigma}^2 = \sum_{j=1}^{n} (\overline{d}_j - \mu)^2 / n$.

5. The method of claim 4, wherein said steps of setting a playout delay value based on said Gaussian delay distribution, determining said probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value further comprise the steps of:

B.2.1: Choosing the initial value of playout delay D as $r*\sigma$;

B.2.2.1: Computing the gap probability using the packet delays in the K^{th} group and new packet delays for I-policy according to $\Pr(gap_1^I) = \lim_{M \to \infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D)$ and

$$\Pr(gap_2^I) = \lim_{M \to \infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D)$$

B.2.2.2: If the value computed at step B.2.2.1 is greater than said predetermined threshold, then executing step B.2.2.3 and otherwise executing said step of playing out said voice packets according to said playout delay;

B.2.2.3: Computing the gap probability for E-policy according to $\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)$ and

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)\right)$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D)\right)$$

B.2.2.4: If the gap probability is less than said predetermined threshold then executing said step of playing out said voice packets according to said playout delay and otherwise adding σ to D and executing step B.2.2.1.

6. The method of claim 1, wherein said step of estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets further comprises the steps of:

C.1.1: Dividing the delays into K groups with each size greater than M.

C.1.2: Computing mean delay $\overline{d}_j$ of talkspurt j, j=1, 2, ..., K, using $\overline{d}_j = \frac{\sum_{i=1}^{m} d_i}{m}$;

C.1.3: Computing the mean delay μ of a series of K groups by $\hat{\mu} = \frac{\sum_{j=1}^{n} \overline{d}_j}{n}$;

C.1.4: Computing the standard deviation σ for the K groups by $\hat{\sigma}^2 = \sum_{j=1}^{n} (\overline{d}_j - \mu)^2 / n$.

7. The method of claim 6, wherein said steps of setting a playout delay value based on said Gaussian delay distribution, determining said probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value further comprise the steps of:

C.2.1: Choosing the initial value of playout delay D as r*σ;

C.2.2.1: Computing the gap probability using the packet delays in the K^{th} group and new packet delays for I-policy according to $\Pr(gap_1^I) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D)$ and

$$\Pr(gap_2^I) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D)$$

C.2.2.2: If the value computed at step C.2.2.1 is greater than said predetermined threshold, then executing step C.2.2.3 and otherwise executing said step of playing out said voice packets according to said playout delay;

C.2.2.3: Computing the gap probability for E-policy according to $\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)$ and

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)\right)$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D)\right)$$

C.2.2.4: If the gap probability is less than said predetermined threshold then executing said step of playing out said voice packets according to said playout delay and otherwise adding σ to D and executing step C.2.2.1.

8. An adaptive playout apparatus for voice packets received over a network, comprising:

estimating means for estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets;

setting means for setting a playout delay value based on said Gaussian delay distribution;

determining means for determining probabilities for single gap and multiple contiguous gaps between said voice packets and for dynamically adjusting said playout delay value as required to satisfy the determined probabilities for single gap and multiple contiguous gaps between said voice packets; and

playout means for playing out said voice packets according to said playout delay.

9. The apparatus of claim 8, wherein said means for estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets further comprises:

means for choosing m largest delays from all voice packets of each talkspurt;

means for, if m is less than a predetermined sample size M for each talkspurt, ignoring said given talkspurt, and otherwise computing the mean delay $\bar{d}_j$ of a talkspurt j, j=1, 2, ..., n as $\overline{d}_j = \frac{\sum_{i=1}^{m} d_i}{m}$;

means for computing the mean delay μ of a series of n talkspurts as $\hat{\mu} = \frac{\sum_{j=1}^{n} \overline{d}_j}{n}$; and

means for computing the standard deviation σ of said n talkspurts as $\hat{\sigma}^2 = \sum_{j=1}^{n}(\bar{d}_j - \mu)^2 / n$.

10. The apparatus of claim 9, wherein said means for setting a playout delay value based on said Gaussian delay distribution, and said means for determining said probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value further comprise:

means for choosing an initial value D of said playout delay as $r*\sigma$;

means for computing the gap probability using the packet delays in an n^{th} talkspurt for I-policy according to

$$\Pr(gap_1^I) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D)$$

and

$$\Pr(gap_2^I) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D)$$

means for, if the computed gap probability is greater than said predetermined threshold, computing the gap probability for E-policy according to

$$\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)$$

and

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D) \right)$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D) \right)$$; and

means for, if the computed gap probability is not less than said predetermined threshold, adding σ to D for the computation of the gap probability using the packet delays.

11. The apparatus of claim 8, wherein said means for estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets further comprises:

means for computing mean delay $\bar{d}_j$ of talkspurt j, j=1, 2, ..., K, using $\bar{d}_j = \frac{\sum_{i=1}^{m} d_i}{m}$ and K = N/M groups;

means for computing the mean delay μ of a series of K groups by $\hat{\mu} = \frac{\sum_{j=1}^{n} \bar{d}_j}{n}$; and

means for computing the standard deviation σ for the K groups by $\hat{\sigma}^2 = \sum_{j=1}^{n} (\bar{d}_j - \mu)^2 / n$.

12. The apparatus of claim 11, wherein said means for setting a playout delay value based on said Gaussian

delay distribution, and said means for determining said probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value further comprise:

means allowing for the choice of the initial value of playout delay D as r*σ;

means for computing the gap probability using the packet delays in the K^{th} group and new packet delays for I-policy according to $\Pr(gap_1^I) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D)$ and

$$\Pr(gap_2^I) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D)$$

means for, if the computed gap probability is greater than said predetermined threshold, computing the gap probability for E-policy according to

$\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)$ and

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D) \right)$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D) \right)$$

means for, if the gap probability for E-policy not is less than said predetermined threshold, adding σ to D for the computation of the gap probability using the packet delays.

13. The apparatus of claim 8, wherein said means for estimating the mean delay and delay variance for said voice packets and in response computing a Gaussian delay distribution for arrival of said voice packets further comprises:

means for dividing the delays into K groups with each size greater than M.

means for computing mean delay $\overline{d}_j$ of talkspurt j, j=1, 2, ..., K, using

$$\overline{d}_j = \frac{\sum_{i=1}^{m} d_i}{m};$$

means for computing the mean delay μ of a series of K groups by

$$\hat{\mu} = \frac{\sum_{i=1}^{n} \overline{d}_j}{n}; \text{ and}$$

means for computing the standard deviation σ for the K groups by

$$\hat{\sigma}^2 = \sum_{j=1}^{n} (\overline{d}_j - \mu)^2 / n.$$

14. The apparatus of claim 13, wherein said means for setting a playout delay value based on said Gaussian delay distribution and said means for determining said probabilities for single gap and multiple contiguous gaps between said voice packets and dynamically adjusting said playout delay value further comprise:

means allowing for the choice of the initial value of playout delay D as r*σ;

means for computing the gap probability using the packet delays in the K^{th} group and new packet delays for I-policy according to $\Pr(gap_1^I) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \Pr(d_i > d_0 + D)$ and

$$\Pr(gap_2^I) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \Pr(d_i > d_0 + D) \Pr(d_{i+1} > d_0 + D)$$

means for, if the computed gap probability is greater than said predetermined threshold, computing the gap probability for E-policy according to

$$\Pr(gap_1^E) = \lim_{M\to\infty} \frac{1}{M-1} \sum_{i=1}^{M-1} \prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D)$$ and

$$\Pr(gap_2^E) = \lim_{M\to\infty} \frac{1}{M-2} \sum_{i=1}^{M-2} \left(\prod_{j=0}^{i-1} \Pr(d_i > d_j + \delta_{0j} D) \right)$$

$$\left(\prod_{j=0}^{i} \Pr(d_{i+1} > d_j + \delta_{0j} D) \right)$$ and

means for, if the gap probability is not less than said predetermined threshold, adding σ to D for the computation of the gap probability using the packet delays.

15. An adaptive playout method for audio packets received over a network, comprising the steps of:

estimating an average delay and delay variance for said audio packets and in response computing a delay distribution for arrival of said audio packets;

setting a playout delay value based on said delay distribution;

determining probabilities for single gap and multiple contiguous gaps between said audio packets and dynamically adjusting said playout delay value as required to satisfy the determined probabilities for single gap and multiple contiguous gaps between said audio packets; and

playing out said audio packets according to said playout delay.

16. An adaptive playout apparatus for audio packets received over a network, comprising:

estimating means for estimating an average delay and delay variance for said audio packets and in response computing a delay distribution for arrival of said audio packets;

setting means for setting a playout delay value based on said delay distribution;

determining means for determining probabilities for single gap and multiple contiguous gaps between said audio packets and for dynamically adjusting said playout delay value as required to satisfy the determined probabilities for single gap and multiple contiguous gaps between said audio packets; and

playout means for playing out said audio packets according to said playout delay.

17. A computer program for controlling a computer to carry out the method according to any one of claims 1 to 7 or 15.

18. A carrier medium carrying the computer program according to claim 15.